

RECEIVED  
CENTRAL FAX CENTER

NOV 22 2005

**Yee &  
Associates, P.C.**4100 Alpha Road  
Suite 1100  
Dallas, Texas 75244Main No. (972) 385-8777  
Facsimile (972) 385-7766**FACSIMILE COVER SHEET**

To: Commissioner for Patents for Examiner Kristina B. Honeycutt Group Art Unit 2178	Facsimile No. 571/273-8300
From: Stephanie Fay Legal Assistant to Betty Formby	No. of Pages Including Cover Sheet: 24
Enclosed herewith: <ul style="list-style-type: none"><li>• Transmittal document; and</li><li>• Appeal Brief.</li></ul>	
Re: Application Serial No. 10/046,999 Attorney Docket No. AUS920010581US1	
Date: Tuesday, November 22, 2005	
Please contact us at (972) 385-8777 if you do not receive all pages indicated above or experience any difficulty in receiving this facsimile.	<i>This Facsimile is intended only for the use of the addressee and, if the addressee is a client or their agent, contains privileged and confidential information. If you are not the intended recipient of this facsimile, you have received this facsimile inadvertently and in error. Any review, dissemination, distribution, or copying is strictly prohibited. If you received this facsimile in error, please notify us by telephone and return the facsimile to us immediately.</i>

**PLEASE CONFIRM RECEIPT OF THIS TRANSMISSION BY  
FAXING A CONFIRMATION TO 972-385-7766.**

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Lakhdir et al.

§ Group Art Unit: 2178

Serial No.: 10/046,999

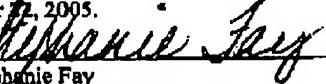
§ Examiner: Honeycutt, Kristina B.

Filed: January 16, 2002

§ Attorney Docket No.: AUS920010581US1

For: Offline Dynamic Web Page  
Generation35525  
PATENT TRADEMARK OFFICE  
CUSTOMER NUMBER**Certificate of Transmission Under 37 C.F.R. § 1.8(a)**

I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (571) 273-8300 on November 22, 2005.

By:   
Stephanie Fay**TRANSMITTAL DOCUMENT**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

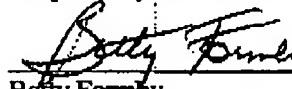
Sir:

ENCLOSED HEREWITH:

- Appeal Brief (37 C.F.R. 41.37).

A fee of \$500.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0447. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to IBM Corporation Deposit Account No. 09-0447. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0447.

Respectfully submitted,

  
Betty Formby  
Registration No. 36,536  
AGENT FOR APPLICANTS

Duke W. Yee  
Registration No. 34,285  
ATTORNEY FOR APPLICANTS

YEE & ASSOCIATES, P.C.  
P.O. Box 802333  
Dallas, Texas 75380  
(972) 385-8777

RECEIVED  
CENTRAL FAX CENTER

Docket No. AUS920010581US1 **PATENT**

NOV 22 2005

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re application of: Lakhdir et al.

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

§

Group Art Unit: 2178

Examiner: Honeycutt, Kristina B.

Serial No. 10/046,999

Filed: January 16, 2002

For: Offline Dynamic Web Page Generation

**Commissioner for Patents**

P.O. Box 1450

Alexandria, VA 22313-1450

Certificate of Transmission Under 37 C.F.R. 1.8(a)

I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (571) 273-8300 on November 22, 2005.

By:

Stephanie Fey

**APPEAL BRIEF (37 C.F.R. 41.37)**

This brief is in furtherance of the Notice of Appeal, filed in this case on September 26, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

11/23/2005 SFELEKE1 00000085 090447 10046999  
01 FC:1402 500.00 DA

(Appeal Brief Page 1 of 22)  
Lakhdir et al. - 10/046,999

**REAL PARTY IN INTEREST**

The real party in interest in this appeal is the following party: International Business Machines Corporation of Armonk, New York.

(Appeal Brief Page 2 of 22)  
Lakhdar et al. - 10/046,999

**RELATED APPEALS AND INTERFERENCES**

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

(Appeal Brief Page 3 of 22)  
Lekhdir et al. - 10/046,999

**STATUS OF CLAIMS**

**A. TOTAL NUMBER OF CLAIMS IN APPLICATION**

Claims in the application are: 1-45

**B. STATUS OF ALL THE CLAIMS IN APPLICATION**

1. Claims canceled: 3, 14-16, 19, 30-32, 35, and 45
2. Claims withdrawn from consideration but not canceled: None
3. Claims pending: 1-2, 4-13, 17-18, 20-29, 33-34, and 36-44
4. Claims allowed: None
5. Claims rejected: 1-2, 4-13, 17-18, 20-29, 33-34, and 36-44
6. Claims objected to: None

**C. CLAIMS ON APPEAL**

The claims on appeal are: 1-2, 4-13, 17-18, 20-29, 33-34, and 36-44

(Appeal Brief Page 4 of 22)  
Lakhdhir et al. - 10/046,999

**STATUS OF AMENDMENTS**

No amendments have been submitted since the Final Office Action was issued.

(Appeal Brief Page 5 of 22)  
Lakhdar et al. - 10/046,999

**SUMMARY OF CLAIMED SUBJECT MATTER****A. CLAIM 1 - INDEPENDENT**

The subject matter of claim 1 is directed to a method of maintaining a web page. The method is shown in the flowchart of Figure 9, discussed in the application on page 18, line 24 through page 19, line 19, and contains the following computer-implemented steps:

at regular intervals or when dynamic content changes, performing the steps listed below, whereby a server responding to a request for dynamic content does not need to render the dynamic content. (The frequency with which the webpage is re-created is discussed on page 15, line 26 through page 16, line 3. It is noted that many of the details of the method are discussed in greater detail during the discussion of the rendering program 402 and various databases 404, 412, 414, 416, and 418 in Figure 4, discussed on page 13, line 24 through page 15, line 25.)

retrieving a source document for said web page. (This step is not explicitly shown, but is implied in step 900, which reads a line of the source document. The database containing the source documents is shown in Figure 4 as item 412, discussed on page 13, lines 15-23.)

locating a plurality of command strings within said source document, wherein each command string of said plurality of command strings includes a respective element type and at least one respective element parameter. (Step 900, which reads a line of source document, and step 902, which determines if the current line is a command string, together locate a command string. These steps are part of a loop that will repeat to locate all of the command strings in the document. Steps 900 and 902 are discussed in the application on page 18, lines 26-28. The content of the command strings is discussed on page 14, line 31 through page 15, line 10.)

retrieving a respective base string corresponding to said respective element type. (Step 910, discussed on page 19, lines 1-4.)

modifying said respective base string according to said at least one respective element parameter to obtain a respective rendered string (Step 912, discussed on page 19, lines 4-5.)

replacing said respective command string in said source document with said respective rendered string. (Step 914, discussed on page 19, lines 5-8.)

after said retrieving, locating, modifying, and replacing steps, saving said source document as a current web page. (This step is not explicitly shown, but is discussed on page 15, lines 19-25, which discloses storing the resulting page in website documents 404 of Figure 4.)

**B. CLAIM 17 - INDEPENDENT**

The subject matter of claim 17 is directed to a computer program product containing instruction that correspond to claim 1. (The computer program product is discussed in the application on page 19, line 20 through page 20, line 6.)

**C. CLAIM 33 - INDEPENDENT**

The subject matter of claim 33 is directed to a data processing system that performs steps corresponding to claim 1. (The data processing system is shown generally in Figures 1 and 2, discussed on page 8, line 2 through page 10, line 21 and more specifically in Figure 4, discussed on page 12, line 28 through page 16, line 3.)

**GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL****A. GROUND OF REJECTION 1 (Claims 1, 2, 4, 6-8, 11-13, 17, 18, 20, 22-24, 27-29, 33, 34, 36, 38-40, 43 and 44)**

Claims 1, 2, 4, 6-8, 11-13, 17, 18, 20, 22-24, 27-29, 33, 34, 36, 38-40, 43 and 44 stand rejected under 35 U.S.C. § 103(a) as obvious over Donohue et al., Method and System for Delivering Documents Customized for a Particular User Over the Internet Using Imbedded Dynamic Content, U.S. Patent No. 5,987,480, November 16, 1999\* (hereinafter “Donohue”) in view of Agrawal et al., Methods and Systems for Partial Page Caching of Dynamically Generated Content, U.S. Patent Publication No. 2002/0004813, January 10, 2002 (hereinafter “Agrawal”).

**B. GROUND OF REJECTION 2 (Claims 5, 21 and 37)**

Claims 5, 21 and 37 stand rejected under 35 U.S.C. § 103(a) as obvious over Donohue in view of Agrawal in further view of Schneider, Method, Product, and Apparatus for Processing a Data Request, U.S. Patent No. 6,760,746, July 6, 2004 (hereinafter “Schneider”).

**C. GROUND OF REJECTION 3 (Claims 9, 10, 25, 26, 41 and 42)**

Claims 9, 10, 25, 26, 41 and 42 stand rejected under 35 U.S.C. § 103(a) as obvious over Donohue in view of Agrawal in further view of Lakritz, Multilingual Electronic Document Translation, Management, and Delivery System, U.S. Patent No. 6,623,529, September 23, 2003 (hereinafter “Lakritz”).

\* This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

ARGUMENT**A. GROUND OF REJECTION 1 (Claims 1-2, 4, 6-8, 11-13, 17-18, 20, 22-24, 27-29, 33-34, 36, 38-40, and 43-44)**

Regarding representative claim 1, the final rejection states:

Regarding independent claim 1, Donohue discloses retrieving a source document for said web page (Fig. 2; col. 8, lines 25-37).

Donohue further discloses locating a plurality of command strings within said source document, wherein each command string of said plurality of command strings includes a respective element type and at least one respective element parameter (Fig. 4; col. 3, lines 51-61; col. 7, lines 45-55; col. 8, lines 15-20, 39-46, 58-67; col. 9, lines 8-15; col. 10, lines 49-55).

Donohue further discloses retrieving a respective base string corresponding to said respective element type (Fig. 4; col. 3, lines 62-65; col. 7, lines 45-57; col. 8, lines 3-9, 58-67; col. 9, lines 8-15; col. 10, lines 34-37).

Donohue further discloses modifying said respective base string according to said at least one respective element parameter to obtain a rendered string (Fig. 4; col. 3, lines 62-67; col. 4, lines 1-4, 45-58; col. 8, lines 58-67; col. 9, lines 8-15, 27-29; col. 10, lines 60-67; col. 11, lines 1-9).

Donohue further discloses replacing the command string the source document with the rendered string (Fig. 4; col. 3, lines 62-67; col. 4, lines 1-4, 45-58; col. 8, lines 58-67; col. 9, lines 8-15, 27-29; col. 10, lines 60-67; col. 11, lines 1-9).

Donohue does not disclose after said retrieving, locating, modifying, and replacing steps, saving said source document as current web page whereby a server responding to a request for dynamic content does not need to render the dynamic content. Agrawal teaches saving a web page so that a server responding to a request does not need to render the dynamic content (p.1, para. 10). It would have been obvious to one of ordinary skill in the art, having the teachings of Donohue and Agrawal before him at the time the invention was made, to modify the current web page taught by Donohue to include saving the web page so that the server does not have to render the page upon request as taught by Agrawal, because saving a web page so that the server doesn't have to render the content, as taught by Agrawal (p.1, para. 10), would allow for quicker display of web pages since the web page only has to be retrieved instead of the dynamic content being rendered in the page.

If the Patent Office does not produce a *prima facie* case of unpatentability, then without more the applicant is entitled to grant of a patent. *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Grabiak*, 769 F.2d 729, 733, 226 U.S.P.Q. 870, 873 (Fed. Cir. 1985). A *prima facie* case of obviousness is established when the teachings of the

prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990).

Claim 1 is not obvious over the combination of **Donohue** and **Agrawal** for at least the following two reasons: (a) neither of these references disclose the feature of "*performing the ... steps*" to produce dynamic web pages "*at regular intervals or when dynamic content changes*" and (b) one of ordinary skill in the art would not use the suggested motivation to combine these two references. These assertions will be discussed separately.

(a) **Performing the steps "at regular intervals or when dynamic content changes"**

The rejection in the final office action does not address the feature of "*at regular intervals or when dynamic content changes, performing the following steps*". Even though this feature was not addressed in the rejection, it is an important feature; ignoring it changes the whole meaning of the claim. As a consequence of performing the steps using this timing, "*a server responding to a request for dynamic content does not need to render the dynamic content*". The rejection interprets that it is the "saving" step which relieves the server from needing to render the dynamic content, referring to **Agrawal**, which states:

[0010] Caching has emerged as the most important solution for reducing the cost of dynamically generating Web pages. The premise underlying caching is that in many cases, the content of dynamically generated pages does not change very frequently, thereby allowing the accessed pages to be cached. The server caches (locally stores for later retrieval) copies of such dynamically generated Web pages. Conventionally, as shown in FIG. 1 at S11, a client sends a Hypertext Transfer Protocol (HTTP) request for a Web page by, for example, clicking on a hyperlink or by specifying a Universal Resource Locator (URL). As shown at S12, the HTTP request is then received by the server, which then checks its cache to see if the response to the request (the entire requested Web page) is stored in the cache, as shown at S13. As called for by S18, if the requested page is indeed present in the cache and if the response page is still valid, the page is extracted from the cache and sent to the client, as shown at S17. If the entire requested page is not present in the cache or is present but invalid (stale), application logic and/or the Web page's script is executed, as shown at S14. Executing the application logic and/or script may require the server to access data sources such as databases or information feeds, as called for by step S15. Only thereafter may the requested Web page be dynamically

generated and sent to the client, as specified in steps S16 and S17. A step of caching the just-generated full page in memory may also be carried out. This mechanism works very well as long as the dynamically generated pages can be cached efficiently (i.e., they are not invalidated very frequently), and the cost of servicing the page from the cache is less than the cost of re-generating the page again. As a result of efficient caching, a Web server uses fewer resources, especially Central Processing Unit resources to service a client's HTTP request. The ability to cache and to efficiently retrieve cached pages improves the server's performance (reduced response time) as well as its capacity to acceptably accommodate a greater number of users than it would otherwise have been able to serve without resorting to caching.

While this limited statement is true, it is not all of the truth. In *Agrawal*, a server that receives a request for dynamic content does not have to render the page IF the page is still in its cache. This will be dependent on many factors, such as the amount of cache available, the number of web pages handled by the server, the frequency of their being requested, etc. Sometimes *Agrawal* will be able to find a page in cache and sometimes it will have to dynamically generate the page. Using the invention recited in claim 1, a server responding to a request for dynamic content does not ever render dynamic content, because all of the steps for creating a web page with dynamic content were performed beforehand. According to the invention recited in claim 1, the steps can be performed "*at regular intervals*" or they can be performed "*when dynamic content changes*", but they are performed at times other than when the page is requested.

*Agrawal* creates a web page and saves it to the cache, but *Agrawal* does not perform all of the steps of creating the web page either "*at regular intervals*" or "*when dynamic content changes*", so this patent does not meet the recited features of representative claim 1.

Likewise, *Donohue* does not show performing the steps to create the web page either "*at regular intervals*" or "*when dynamic content changes*". Rather, this patent notes, "*in response to such a request, the server computer selects one of the document templates corresponding to the desired document, populates the document template with content ... and delivers the populated document to the client computer*"<sup>1</sup>.

Thus, neither *Agrawal* nor *Donohue* disclose that the steps of creating the dynamic web page are being performed either "*at regular intervals*" or "*when dynamic content changes*". Both

---

<sup>1</sup> *Donohue, abstract*

of these patents create the dynamic web page in response to a request for the page, not as a separate act that creates the page before it is requested.

**(b) Motivation to combine**

The rejection offers, as a motivation to combine, because it "would allow for quicker display of web pages since the web page only has to be retrieved instead of the dynamic content being rendered in the page". However, it is submitted that the mechanism of caching does not fit well with the customization that is the aim of Donohue. As Agrawal notes in the excerpt cited above, "*This mechanism [caching] works very well as long as the dynamically generated pages can be cached efficiently (i.e., they are not invalidated very frequently)*". However, a major focus of Donohue is customizing web pages; this patent notes, "*In preferred embodiments, the customized content is provided based on the identity of the user operating the client computer so that the document is individualized to the user's interests and needs.*"<sup>2</sup> A high level of customization in the production of web pages, such as customizing for individuals, would lead to a greater inability to reuse web pages. A page that is customized for a first user would not necessarily be identical to a page customized for a second user, even if the two users were requesting the same information. In fact, as Donohue's goal of customizing is more clearly met, there would be a proportional decrease in the ability to utilize the caching of Agrawal to advantage. Therefore, one of ordinary skill in the art would not be motivated to combine these references to reach the claimed invention.

Appellants have shown two reasons why claim 1 is not obvious over the combination of Donohue and Agrawal. This rejection is overcome for claim 1. The rejection is also overcome for claims 17 and 33, which are rejected for similar reasons as claim 1. Additionally, the remaining claims in this rejection as dependent on one of claims 1, 17, or 33 and so their rejections are also overcome.

**B. GROUND OF REJECTION 2 (Claims 5, 21 and 37)**

This rejection is not argued separately; each of these claims is dependent on one of the claims in the first ground of rejection. These claims can stand or fall with their independent claims.

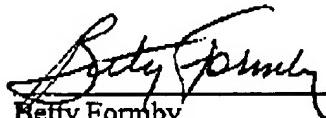
---

<sup>2</sup> Donohue, column 4, lines 16-19

**C. GROUND OF REJECTION 3 (Claims 9, 10, 25, 26, 41 and 42)**

This rejection is not argued separately; each of these claims is dependent on one of the claims in the first ground of rejection. These claims can stand or fall with their independent claims.

The Board of Appeals is requested to reverse the outstanding rejection and to indicate all claims to be allowable.



Betty Formby  
Reg. No. 36,536  
YEE & ASSOCIATES, P.C.  
PO Box 802333  
Dallas, TX 75380  
(972) 385-8777

(Appeal Brief Page 13 of 22)  
Lakhdir et al. - 10/046,999

**CLAIMS APPENDIX**

The text of the claims involved in the appeal are:

1. A method of maintaining a web page, comprising the computer-implemented steps:
  - at regular intervals or when dynamic content changes, performing the following steps:
    - retrieving a source document for said web page;
    - locating a plurality of command strings within said source document, wherein each command string of said plurality of command strings includes a respective element type and at least one respective element parameter;
    - retrieving a respective base string corresponding to said respective element type;
    - modifying said respective base string according to said at least one respective element parameter to obtain a respective rendered string;
    - replacing said respective command string in said source document with said respective rendered string;
  - after said retrieving, locating, modifying, and replacing steps, saving said source document as a current web page;
  - whereby a server responding to a request for dynamic content does not need to render the dynamic content.
2. The method of claim 1, wherein the base string is retrieved from a data structure.
4. The method of claim 1, wherein the at least one element parameter includes one of a name, a value, a description, a number of columns, or a format modifier.

5. The method of claim 1, wherein the at least one element parameter includes a database domain.
6. The method of claim 1, wherein the element type is one of checkbox, selection, radiobutton, textarea, button, heading, or title.
7. The method of claim 1, wherein the base string includes tags written in a structure markup language.
8. The method of claim 7, wherein the structured markup language is one of Hypertext Markup Language (HTML), Extensible Markup Language (XML), Wireless Markup Language (WML), or Standard Generalized Markup Language (SGML).
9. The method of claim 1, wherein the base string includes an embedded script.
10. The method of claim 9, wherein the embedded script is one of a client-side script or a server-side script.
11. The method of claim 1, wherein modifying the base string according to the parameters to obtain a rendered string includes replacing a substring within the base string with one of the element parameters.
12. The method of claim 1, wherein modifying the base string according to the parameters to obtain a rendered string includes:

using one of the element parameters to retrieve a replacement substring from a database;  
and

replacing a substring within the base string with the replacement substring.

13. The method of claim 1, wherein the method is performed in a rendering program.
17. A computer program product in a computer-readable medium for maintaining a web page, comprising instructions for:
  - at regular intervals or when dynamic content changes, performing the following steps:
    - retrieving a source document for said web page;
    - locating a plurality of command strings within said source document, wherein each command string of said plurality of command strings includes a respective element type and at least one respective element parameter;
    - retrieving a respective base string corresponding to said respective element type;
    - modifying said respective base string according to said at least one respective element parameter to obtain a respective rendered string;
    - replacing said respective command string in said source document with said respective rendered string;
    - after said retrieving, locating, modifying, and replacing steps, saving said source document as a current web page;
  - whereby a server responding to a request for dynamic content does not need to render the dynamic content.

18. The computer program product of claim 17, wherein the base string is retrieved from a data structure.
20. The computer program product of claim 17, wherein the at least one element parameter includes one of a name, a value, a description, a number of columns, or a format modifier.
21. The computer program product of claim 17, wherein the at least one element parameter includes a database domain.
22. The computer program product of claim 17, wherein the element type is one of checkbox, selection, radiobutton, textarea, button, heading, or title.
23. The computer program product of claim 17, wherein the base string includes tags written in a structure markup language.
24. The computer program product of claim 23, wherein the structured markup language is one of Hypertext Markup Language (HTML), Extensible Markup Language (XML), Wireless Markup Language (WML), or Standard Generalized Markup Language (SGML).
25. The computer program product of claim 17, wherein the base string includes an embedded script.
26. The computer program product of claim 25, wherein the embedded script is one of a client-side script or a server-side script.

27. The computer program product of claim 17, wherein modifying the base string according to the parameters to obtain a rendered string includes replacing a substring within the base string with one of the element parameters.

28. The computer program product of claim 17, wherein modifying the base string according to the parameters to obtain a rendered string includes:

using one of the element parameters to retrieve a replacement substring from a database;  
and

replacing a substring within the base string with the replacement substring.

29. The computer program product of claim 17, wherein said computer program is a rendering program.

33. (Currently amended) A data processing system for maintaining a web page, comprising:  
a bus system;

a processing unit connected to the bus system, wherein the processing unit includes at least one processor;

at regular intervals or when dynamic content changes, performing the following steps:

retrieving a source document for said web page;

locating a plurality of command strings within said source document, wherein

each command string of said plurality of command strings includes a respective element type and at least one respective element parameter;

retrieving a respective base string corresponding to said respective element type;

- modifying said respective base string according to said at least one respective element parameter to obtain a respective rendered string;
- replacing said respective command string in said source document with said respective rendered string;
- after said retrieving, locating, modifying, and replacing steps, saving said source document as a current web page;
- whereby a server responding to a request for dynamic content does not need to render the dynamic content.
34. The data processing system of claim 33, wherein the base string is retrieved from a data structure.
36. The data processing system of claim 33, wherein the at least one element parameter includes one of a name, a value, a description, a number of columns, or a format modifier.
37. The data processing system of claim 33, wherein the at least one element parameter includes a database domain.
38. The data processing system of claim 33, wherein the element type is one of checkbox, selection, radiobutton, textarea, button, heading, or title.
39. The data processing system of claim 33, wherein the base string includes tags written in a structure markup language.

40. The data processing system of claim 39, wherein the structured markup language is one of Hypertext Markup Language (HTML), Extensible Markup Language (XML), Wireless Markup Language (WML), or Standard Generalized Markup Language (SGML).

41. The data processing system of claim 33, wherein the base string includes an embedded script.

42. The data processing system of claim 41, wherein the embedded script is one of a client-side script or a server-side script.

43. The data processing system of claim 33, wherein modifying the base string according to the parameters to obtain a rendered string includes replacing a substring within the base string with one of the element parameters.

44. The data processing system of claim 33, wherein modifying the base string according to the parameters to obtain a rendered string includes:

using one of the element parameters to retrieve a replacement substring from a database;  
and

replacing a substring within the base string with the replacement substring.

(Appeal Brief Page 20 of 22)  
Lakhdhir et al. - 10/046,999

**EVIDENCE APPENDIX**

There is no evidence to be presented.

(Appeal Brief Page 21 of 22)  
Lakhdir et al. - 10/046,999

**RELATED PROCEEDINGS APPENDIX**

There are no related proceedings.

(Appeal Brief Page 22 of 22)  
Lekhdar et al. - 10/046,999